IN THE CLAIMS:

Please AMEND the claims in accordance with the following:

1. (Currently Amended) A computer which performs parallel processing of a plurality of programs in a time-division fashion, comprising:

a memory;

an instruction fetch unit configured to fetch instructions from said memory;

hardware resources divided into a plurality of areas, the hardware resources being used in common by a plurality of programs ;

an evacuation unit configured to record identification ~~information~~ data identifying a first program as relating to information stored in a given area of the plurality of areas if the given area is used for execution of the first program, and to evacuate information stored in ~~a first~~the given area ~~of said plurality of areas~~ if the ~~first~~ given area used by the first program is necessary for execution of a second program ~~and is being used for execution of the first program~~, said information being evacuated to a portion of said memory that corresponds to the first program; and

a restoration unit configured to restore, ~~from the memory~~ to ~~the~~ a first area of the plurality of areas, a first part of information necessary for execution of the second program, to mark the first area as a usable area while marking areas other than the first area as unusable areas, to restore, ~~from the memory~~ to a second area of the plurality of areas, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and~~ to restore to the first area the ~~evacuated~~ information ~~to~~ evacuated from the first area based on the identification ~~information~~ data when the second program comes to a halt or to an end, and to determine based on the identification data whether to restore to the second area the information evacuated from the second area.

2. (Currently Amended) The computer as claimed in claim 1, further comprising an interruption unit which brings about interruption processing if the ~~first~~ given area is necessary for execution of ~~a~~ the second program ~~and is being used for execution of the first program~~, wherein said evacuation unit operates as part of the interruption processing to record the identification ~~information~~ data and to evacuate the information stored in the ~~first~~ given area.

3. (Currently Amended) A computer which performs parallel processing of a plurality of programs in a time-division fashion, comprising:

a memory;

an instruction fetch unit configured to fetch instructions from said memory;

hardware resources divided into a plurality of areas, the hardware resources being used in common by a plurality of programs;

an evacuation unit configured to record identification ~~information~~ data identifying a first program as relating to information stored in a given area of the plurality of areas if the given area is used for execution of the first program, and to evacuate information stored in ~~a first~~the given area ~~of said plurality of areas~~ if the ~~first~~ given area used by the first program and a second area of said plurality of areas are necessary for execution of a second program ~~and are being used for execution of the first program~~, said evacuation unit subsequently evacuating information stored in the second area when use of the second area becomes actually necessary for execution of the second program, said information being evacuated to a portion of said memory that corresponds to the first program; and

a restoration unit configured to restore, ~~from the memory~~ to ~~the~~ a first area of the plurality of areas, a first part of information necessary for execution of the second program, to mark the first area as a usable area while marking areas other than the first area as unusable areas, to restore, ~~from the memory~~ to the second area, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and~~ to restore to the first area the ~~evacuated~~ information ~~to~~ evacuated from the first and second areas based on the identification ~~information~~ data when the second program comes to a halt or to an end, and to determine based on the identification data whether to restore to the second area the information evacuated from the second area.

4. (Currently Amended) A method of controlling a computer which performs parallel processing of a plurality of programs in a time-division fashion, said computer having a memory and an instruction fetch unit configured to fetch instructions from said memory, comprising:

providing hardware resources divided into a plurality of areas, the hardware resources being used in common by a plurality of programs;

recording identification ~~information~~ data identifying a first program as relating to information stored in a given area of the plurality of areas if the given area is used for execution of the first program, and evacuating information stored in ~~a first~~the given area ~~of said plurality of areas~~ if the ~~first~~ given area used by the first program is necessary for execution of a second program ~~and is being used for execution of the first program~~, said information being evacuated

to a portion of said memory that corresponds to the first program; and

restoring ~~from the memory~~ to ~~the~~ a first area of the plurality of areas, a first part of information necessary for execution of the second program, to mark the first area as a usable area while marking areas other than the first area as unusable areas, ~~to restore, from the memory~~ restoring to a second area of the plurality of areas, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and to restore~~restoring to the first area the ~~evacuated~~ information ~~to~~ evacuated from the first area based on the identification ~~information~~ data when the second program comes to a halt or to an end, and determining based on the identification data whether to restore to the second area the information evacuated from the second area.

5. (Currently Amended) A method of controlling a computer which performs parallel processing of a plurality of programs in a time-division fashion, said computer having a memory and an instruction fetch unit configured to fetch instructions from said memory, comprising:

providing hardware resources divided into a plurality of areas, the hardware resources being used in common by a plurality of programs;

recording identification ~~information~~ data identifying a first program as relating to information stored in a given area of the plurality of areas if the given area is used for execution of the first program, and evacuating information stored in ~~a first~~the given area ~~of said plurality of areas~~ if the ~~first~~ given area and a second area of said plurality of areas are necessary for execution of a second program ~~and are being used for execution of the first program~~, followed by subsequently evacuating information stored in the second area when use of the second area becomes actually necessary for execution of the second program, said information being evacuated to a portion of said memory that corresponds to the first program; and

restoring ~~from the memory~~ to ~~the~~ a first area of the plurality of areas, a first part of information necessary for execution of the second program, to mark the first area as a usable area while marking areas other than the first area as unusable areas, ~~to restore,~~restoring ~~from the memory~~ to the second area, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and to restore~~restoring to the first area the ~~evacuated~~ information ~~to~~ evacuated from the first and second areas based on the identification ~~information~~ data when the second program comes to a halt or to an end, and determining based on the identification data whether to restore to the second area the information evacuated from the second area.

6. (Cancelled)

7. (Cancelled)

8. (Cancelled)

9. (Cancelled)

10. (Cancelled)

11. (Cancelled)

12. (Cancelled)

13. (Cancelled)

14. (Cancelled)

15. (Cancelled)

16. (Cancelled)

17. (Cancelled)

18. (Currently Amended) A computer for parallel processing, comprising:

an evacuation unit configured to record identification ~~information~~ data identifying a first program as relating to information stored in a given area of a plurality of areas of a hardware resource being used in parallel by at least two of a plurality of programs, if the given area is used for execution of the first program, and to evacuate information stored in ~~a first~~the given area ~~of a plurality of areas of a hardware resource being used in parallel by at least two of a plurality of programs,~~ if the given area used by the first program is necessary for execution of a second program, and evacuates the information to a portion of a memory that corresponds to the first program; and

a restoration unit configured to restore, ~~from the memory~~ to ~~the~~ a first area of the plurality of areas, a first part of information necessary for execution of the second program, to mark the first area as a usable area while marking areas other than the first area as unusable areas, to restore, ~~from the memory~~ to a second area of the plurality of areas, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and~~ to restore to the first area the ~~evacuated~~ information evacuated from ~~to~~ the first area based on the identification ~~information~~ data when the second program comes to a halt or to an end, and to determine based on the identification data whether to restore to the second area the information evacuated from the second area.

19. (Currently Amended) A method for parallel processing comprising:

recording identification ~~information~~ data identifying a first program as relating to information stored in a ~~first~~ given area of ~~said~~ a plurality of areas if the given area is used for execution of the first program, said first area being used to run at least two of a plurality of programs in parallel;

evacuating information stored in the given area to a memory if the ~~first~~ given area is necessary for execution of a second program; and

restoring ~~from the memory~~ to ~~the~~ a first area of the plurality of areas, a first part of information necessary for execution of the second program, to mark the first area as a usable area while marking areas other than the first area as unusable areas, to restore~~, from the memory~~ to ~~the~~ a second area of the plurality of areas, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and~~ to restore to the first area the ~~evacuated~~ information evacuated from ~~to~~ the first area based on the identification ~~information~~ data when the second program comes to a halt or to an end, and to determine based on the identification data whether to restore to the second area the information evacuated from the second area.

20. (Currently Amended) A method, comprising:

evacuating information associated with an executing first program and identified by a first program identifier where the information is located in a ~~first~~ given area of a plurality of areas of a processor and is evacuated to outside the processor when the given area is necessary for execution of a second program; and

restoring from outside the processor to the first area, a first part of information necessary

for execution of the second program, to mark the first area as a usable area while marking an other area other than the first area as an unusable area, to restore, from outside the processor to a second area of the plurality of areas, a second part of the information necessary for execution of the second program if execution of the second program needs to use an area that is marked as an unusable area, ~~and~~ to restore to the first area the ~~evacuated~~ information evacuated from ~~to~~ the first area using the identifier when the second program ends execution, and to determine based on the identifier whether to restore to the second area the information evacuated from the second area.